COSC 4330 FIRST MIDTERM JUNE 6, 2002

This exam is closed book. You can have one page of notes. UH expels cheaters.

1. *For each of the statements below, indicate in one sentence whether the statement is true or false* (2 points), ***and why*** (3 points).

a) An OS that allows people to reboot the system from a floppy drive without entering first a password is not secure.

TRUE, we could reboot the system with a doctored OS that would allow us full access to all files.

b) UNIX was the first system to have a graphical user interface.

FALSE, the first commercially successful system with a graphical interface was the Apple Macintosh..

c) Disk drives are half a million times slower than main memory.

TRUE, memory access times are measured in anonseconds while disk access times ae measured in milliseconds..

d) **`execve()`** system calls are often followed by a **`fork()`** system call.

FALSE, it is the other way around: fork() system calls are often followed by a execve() system call.

e) ***Delaying writes*** generally is a good idea**.**

TRUE, it reduces the number of disk writes by lumping together several writes.

f) Most modern operating systems have a ***layered kernel.***

FALSE, its is too difficult to partition a kernel into meaningful layers that can be stacked on the top of each other.

2. Compare the respective advantages and disadvantages of **user-level** and **kernel-supported** threads in terms of (a) ease of installation, (b) speed, (c) application to multiprocessor architectures and (d) ease of use. (4×5 points)

a) User-level threads are easier to install than kernel-supported threads as they do not require any modification to the kernel.

b) User-level threads offer faster switches between threads sharing the same address space than kernel-supported threads.

c) Kernel-supported threads are the only ones allowing one program to run on several processing unites at the same time.

d) Kernel-supported threads are easier to use as the programmer can use blocking system calls

3. Complete the following fragment of code to ensure that the standard error of the process is redirected to the file `error_file`. (2×5 points)

```
int fd;
fd = open("error_file", O_WRONLY, 0600);
close(2)_____________
dup(fd)____________________
close(fd);
```

4. ***Advantage and disadvantages:*** *you will get no credit if you answer mentions a disadvantage when an advantage is asked and vice versa.* (4×5 points)

a) What is the major advantage of ***monolithic kernels*** over ***microkernels****?*

They are faster because they require two context switches per system call instead of four.

b) What is the major disadvantage of ***master slave organizations*** for multiprocessor operating systems over ***symmetric organizations***?

They require are kernel tasks to be performed on a single processor that can often become a bottleneck.

c) What is the major disadvantage of ***not having memory protection***?

User processes could modify the kernel in order to disable its file access control mechanisms.

d) What would be the major disadvantage of ***single-threaded file server****?*

It would not be able to process as many requests per second as a multi-threaded file server.

5. What will normally bring a process:

a) from the ***running*** state to the ***waiting*** state? (5 points)

The process does a system call.

b) from the ***waiting*** state to the ***waiting_suspended*** state? (5points)

The process has been in the waiting state for a long time and it is swaaped out to make more space in main memory..

6. What is happening when a UNIX process issues a `wait()` system call and all its child processes have already terminated? Will the process wait forever or does UNIX offer a better solution? (10 points)

The wait() will immediately return because UNIX keeps in its process table all processes that have terminated as long as their parent processes have not performed a wait() on them.